The Parameterized Complexity of Guarding Almost Convex Polygons

Akanksha Agrawal
Ben-Gurion University, Beer-Sheba, Israel
agrawal@post.bgu.ac.il

Kristine V. K. Knudsen
University of Bergen, Bergen, Norway
kristine.knudsen@ii.uib.no

Daniel Lokshtanov
University of California, Santa Barbara, CA, USA
daniello@ucsb.edu

Saket Saurabh
The Institute of Mathematical Sciences, HBNI, Chennai, India
saket@imsc.res.in

Meirav Zehavi
Ben-Gurion University, Beer-Sheba, Israel
meiravze@bgu.ac.il

Abstract

The Art Gallery problem is a fundamental visibility problem in Computational Geometry. The input consists of a simple polygon P, (possibly infinite) sets G and C of points within P, and an integer k; the task is to decide if at most k guards can be placed on points in G so that every point in C is visible to at least one guard. In the classic formulation of Art Gallery, G and C consist of all the points within P. Other well-known variants restrict G and C to consist either of all the points on the boundary of P or of all the vertices of P. Recently, three new important discoveries were made: the above mentioned variants of Art Gallery are all W[1]-hard with respect to k [Bonnet and Miltzow, ESA'16], the classic variant has an $\mathcal{O}(\log k)$-approximation algorithm [Bonnet and Miltzow, SoCG'17], and it may require irrational guards [Abrahamsen et al., SoCG'17]. Building upon the third result, the classic variant and the case where G consists only of all the points on the boundary of P were both shown to be $\exists\mathbb{R}$-complete [Abrahamsen et al., STOC'18]. Even when both G and C consist only of all the points on the boundary of P, the problem is not known to be in NP.

Given the first discovery, the following question was posed by Giannopoulos [Lorentz Center Workshop, 2016]: Is Art Gallery FPT with respect to r, the number of reflex vertices? In light of the developments above, we focus on the variant where G and C consist of all the vertices of P, called Vertex-Vertex Art Gallery. Apart from being a variant of Art Gallery, this case can also be viewed as the classic Dominating Set problem in the visibility graph of a polygon. In this article, we show that the answer to the question by Giannopoulos is *positive*: Vertex-Vertex Art Gallery is solvable in time $r^{\mathcal{O}(r^2)}n^{\mathcal{O}(1)}$. Furthermore, our approach extends to assert that Vertex-Boundary Art Gallery and Boundary-Vertex Art Gallery are both FPT as well. To this end, we utilize structural properties of "almost convex polygons" to present a two-stage reduction from Vertex-Vertex Art Gallery to a new constraint satisfaction problem (whose solution is also provided in this paper) where constraints have arity 2 and involve monotone functions.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability; Theory of computation → Computational geometry

Keywords and phrases Art Gallery, Reflex vertices, Monotone 2-CSP, Parameterized Complexity, Fixed Parameter Tractability

Digital Object Identifier 10.4230/LIPIcs.SoCG.2020.3

Related Version A full version of this paper is available at `https://arxiv.org/abs/2003.07793`.

36th International Symposium on Computational Geometry (SoCG 2020).
Editors: Sergio Cabello and Danny Z. Chen; Article No. 3; pp. 3:1–3:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Funding *Akanksha Agrawal*: the PBC Fellowship Program for Outstanding Post-Doctoral Researchers from China and India.
Daniel Lokshtanov: European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (no. 715744), and United States – Israel Binational Science Foundation (no. 2018302).
Saket Saurabh: European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (no. 819416), and Swarnajayanti Fellowship (no. DST/SJF/MSA01/2017-18).
Meirav Zehavi: Israel Science Foundation grant no. 1176/18, and United States – Israel Binational Science Foundation (no. 2018302).

Acknowledgements We thank anonymous reviewers for helpful comments that improved and simplified the paper.

1 Introduction

Given a *simple* polygon P on n vertices, two points x and y within P are *visible* to each other if the line segment between x and y is contained in P. Accordingly, a set S of points within P is said to *guard* another set Q of points within P if, for every point $q \in Q$, there is some point $s \in S$ such that q and s are visible to each other. The computational problem that arises from this notion is loosely termed the ART GALLERY problem. In its general formulation, the input consists of a simple polygon P, possibly infinite sets G and C of points within P, and a non-negative integer k. The task is to decide whether at most k guards can be placed on points in G so that every point in C is visible to at least one guard. The most well-known cases of ART GALLERY are identified as follows: the X-Y ART GALLERY problem is the ART GALLERY problem where G is the set of all points within P (if X=POINT), all boundary points of P (if X=BOUNDARY), or all vertices of P (if X=VERTEX), and C is defined analogously with respect to Y. The classic variant of ART GALLERY is the POINT-POINT ART GALLERY problem. Nevertheless, all variants where X=VERTEX or Y=POINT received attention in the literature.[1] In particular, VERTEX-VERTEX ART GALLERY is equivalent to the classic DOMINATING SET problem in the visibility graph of a polygon.

ART GALLERY is a fundamental visibility problem in Discrete and Computational Geometry, which was extensively studied from both combinatorial and algorithmic viewpoints. The problem was first proposed by Victor Klee in 1973, which prompted a flurry of results [15, page 1]. The main combinatorial question posed by Klee was *how many guards are sufficient to see every point of the interior of an n-vertex simple polygon?* Chvátal [6] showed in 1975 that $\lfloor \frac{n}{3} \rfloor$ guards are always sufficient and sometimes necessary for any n-vertex simple polygon (see [8] for a simpler proof by Fisk). After this, many variants of ART GALLERY, based on different definitions of visibility, restricted classes of polygons, different shapes of guards, and mobility of guards, have been defined and analyzed. A book [15] and several extensive surveys and book chapters were dedicated to ART GALLERY and its variants (see, e.g., [7, 18, 19]). In this article, our main proof states that VERTEX-VERTEX ART GALLERY is *fixed-parameter tractable (*FPT*)* parameterized by r, the number of reflex vertices of P. Additionally, we show that both VERTEX-BOUNDARY ART GALLERY and BOUNDARY-VERTEX ART GALLERY are FPT with respect to the number of reflex vertices as well.

[1] The X-Y ART GALLERY problem, for any X,Y $\in$ {POINT, BOUNDARY, VERTEX}, is often loosely termed the ART GALLERY problem. For example, in the survey of open problems by Ghosh and Goswami [9], the term ART GALLERY problem refers to the VERTEX-VERTEX ART GALLERY problem.

1.1 Background: Related Algorithmic Works

We focus only on algorithmic works on X-Y Art Gallery for X,Y∈ {Point,Boundary, Vertex}. (The discussions regarding known approximation and exact algorithms can be found in the full version [4] of the paper.)

Hardness. In 1983, O'Rourke and Supowit [16] proved that Point-Point Art Gallery is NP-hard if the polygon can contain holes. The requirement to allow holes was lifted shortly afterwards [3]. In 1986, Lee and Lin [12] showed that Vertex-Point Art Gallery is NP-hard. This result extends to Vertex-Vertex Art Gallery and Vertex-Boundary Art Gallery. Later, numerous other restricted cases were shown to be NP-hard as well. For example, NP-hardness was established for orthogonal polygons by Katz and Roisman [11] and Schuchardt and Hecker [17]. We remark that the reductions that show that X-Y Art Gallery (for X,Y ∈ {Point, Boundary, Vertex}) is NP-hard also imply that these cases cannot be solved in time $2^{o(n)}$ under the Exponential-Time Hypothesis (ETH).

While it has long been known that even very restricted cases of Art Gallery are NP-hard, the inclusion of X-Y Art Gallery, for X,Y ∈ {Point, Boundary}, in NP remained open. (When X=Vertex, the problem is clearly in NP.) In 2017, Abrahamsen et al. [1] began to reveal the reasons behind this discrepancy for the Point-Point Art Gallery problem: they showed that *exact* solutions to this problem sometimes require placement of guards on points with *irrational* coordinates. Shortly afterwards, they extended this discovery to prove that Point-Point Art Gallery and Boundary-Point Art Gallery are $\exists\mathbb{R}$-complete [2]. Roughly speaking, this result means that *(i)* any system of polynomial equations over the real numbers can be encoded as an instance of Point/Boundary-Point Art Gallery, and *(ii)* these problems are not in the complexity class NP unless NP = $\exists\mathbb{R}$.

Parameterized Complexity. Two years ago, Bonnet and Miltzow [5] showed that Vertex-Point Art Gallery and Point-Point Art Gallery are W[1]-hard with respect to the *solution size*, k. With straightforward adaptations, their results extend to most of the known variants of the problem, including Vertex-Vertex Art Gallery. Thus, *the classic parameterization by solution size leads to a dead-end.* However, this does not rule out the existence of FPT algorithms for non-trivial structural parametrizations. We refer to the nice surveys by Niedermeier on the art of parameterizations [13, 14].

1.2 Giannopoulos's Parameterization and Our Contribution

In light of the W[1]-hardness result by Bonnet and Miltzow [5], Giannopoulos [10] proposed to parameterize the Art Gallery problem by the number r of reflex vertices of the input polygon P. Specifically, Giannopoulos [10] posed the following open problem: *"Guarding simple polygons has been recently shown to be* W[1]*-hard w.r.t. the number of (vertex or edge) guards. Is the problem* FPT *w.r.t. the number of reflex vertices of the polygon?"* The motivation behind this proposal is encapsulated by the following well-known proposition, see [15, Sections 2.5-2.6].

▶ **Proposition 1** (Folklore). *For any polygon P, the set of reflex vertices of P guards the set of all points within P.*

That is, the minimum number k of guards needed (for any of the cases of Art Gallery) is upper bounded by the number of reflex vertices r. Clearly, k can be arbitrarily smaller than r (see Fig. 1). Our main result is that the Vertex-Vertex Art Gallery problem is FPT parameterized by r. This implies that guarding the vertex set of "almost convex polygons" is easy. In particular, whenever $r^2 \log r = \mathcal{O}(\log n)$, the problem is solvable in polynomial time.

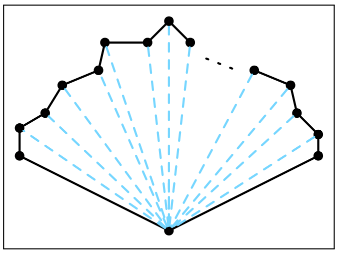

Figure 1 The solution size $k = 1$, yet the number of reflex vertices r is arbitrarily large.

▶ **Theorem 2.** *Vertex-Vertex Art Gallery is* FPT *parameterized by r, the number of reflex vertices. In particular, it admits an algorithm with running time $r^{\mathcal{O}(r^2)}n^{\mathcal{O}(1)}$.*

A few remarks are in place. First, our result extends (with straightforward adaptation) to the most general discrete annotated case of Art Gallery where G and C are each a subset of the vertex set of the polygon, which can include points where the interior angle is of 180 degrees. Consequently, a simple discretization procedure shows that Vertex-Boundary Art Gallery and Boundary-Vertex Art Gallery are both FPT parameterized by r as well. However, we do not know how to handle Vertex-Point Art Gallery and Point-Vertex Art Gallery; determining whether these variants are FPT with respect to r remains open. Second, for variants where both X $\neq$ Vertex and Y $\neq$ Vertex, the design of *exact* algorithms poses extremely difficult challenges. As discussed earlier, these cases are not even known to be in NP; in particular, Point-Point Art gallery is $\exists\mathbb{R}$-hard [2]. Moreover, there is only one known exact algorithm that resolves these cases and it employs extremely powerful machinery (as a black box), not known to be avoidable. Third, note that our result is among very few *positive* results that concern *optimal* solutions to (any case of) Art Gallery.

Along the way to establish our main result, we prove that a constraint satisfaction problem called Monotone 2-CSP is solvable in polynomial time. This result might be of independent interest. Informally, in Monotone 2-CSP, we are given k variables and m constraints. Each constraint is of the form $[x \,\mathsf{sign}\, f(x')]$ where x and x' are variables, $\mathsf{sign} \in \{\leq, \geq\}$, and f is a *monotone* function. The objective is to assign an integer from $\{0, 1, \ldots, N\}$ to each variable so that all of the constraints will be satisfied. For this problem, we develop a surprisingly simple algorithm based on a reduction to 2-CNF-SAT.

▶ **Theorem 3** (♠[2])**.** *Monotone 2-CSP is solvable in polynomial time.*

The main technical component of our work is an exponential-time reduction that creates an exponential (in r) number of instances of Monotone 2-CSP so that the original instance is a Yes-instance if and only if at least one of the instances of Monotone 2-CSP is a Yes-instance. Our reduction is done in two stages due to its structural complexity. In the first stage of the reduction, we aim to make "guesses" that determine the relations between the "elements" of the problem (that are the "critical" visibility relations in our case) and thereby elucidate and further binarize them (which, in our case, is required to impose order on guards). This part requires exponential time (given that there are exponentially many guesses) and captures the "NP-hardness" of the problem. Then, the second stage of the reduction is to translate each guess into an instance of Monotone 2-CSP. This part, while

[2] Details of the results marked with ♠ can be found in the full version of the paper [4].

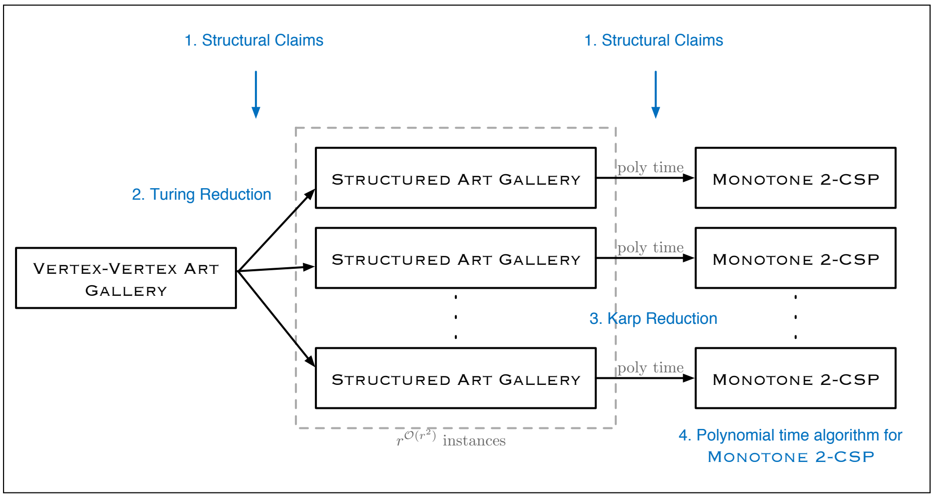

Figure 2 The four components of our proof.

requiring polynomial time, relies on a highly non-trivial problem-specific insight – specifically, here we need to assert that the relations considered earlier can be encoded by constraints that are not only binary, but that the functions they involve are *monotone*. We strongly believe that our approach can be proven fruitful to resolve the parameterized complexity of other problems of discrete geometric flavour.

1.3 Our Methods and Preliminaries

Our Methods. The proof of Theorem 2 consists of four components (see Fig. 2). The first component (in Section 2.1) establishes several structural claims regarding monotone properties of visibility in polygons. Informally, we order the vertices of the polygon according to their appearance on the boundary, and consider each sequence between two reflex vertices to be a "convex region". Then, we argue that for every pair of convex regions, as we "move along" one of them, the (index of the) first vertex in the other region that we see either never becomes smaller or never becomes larger. Symmetrically, this claim also holds for the last visible vertices that we encounter. In addition, we argue that if a vertex sees some two vertices in a convex region, then it also sees all vertices in between these two vertices.

Our second component (in Section 2.2) is a Turing reduction to an intermediate problem that we term Structured Art Gallery. Roughly speaking, in this problem, each convex region "announces" how many guards it will contain, and how many guards are necessary to see it completely. In addition, it announces that a prefix of the sequence that forms this region will be guarded by, say, "the i^{th} guard to be placed on region C", then the following subsequence will be guarded by, say, "the j^{th} guard to be placed on region C'", and so on, until it announces how a suffix of it is to be guarded. We stress that the identity of what is "the i^{th} guard to be placed on region C", or what is "the j^{th} guard to be placed on region C'", are of course not known, and should be discovered. Moreover, even the division into subsequences is not known. In Structured Art Gallery, we only focus on solutions that are of the above form. We utilize our second component not only to impose these additional conditions, but also to begin the transition from the usage of visibility-based conditions to function-based constraints. Specifically, functions called first and last will encode, for any vertex v and convex region C, the first and last vertices in C visible to v. To argue that such simple functions encode all necessary information concerning visibility, we make use of the structural claims established earlier.

Our third component (in Section 2.3) is a Karp reduction from Structured Art Gallery to the constraint satisfaction problem, Monotone 2-CSP, discussed in Section 1.2. This is the part of the proof that most critically relies on all of the structural claims established earlier. Here, we need to translate the constraints imposed by Structured Art Gallery into constraints that comply with the very restricted form of an instance of Monotone 2-CSP, namely, being monotone and involving only two variables. We remark that if one removes the requirement of monotonicity, or allows each constraint to consist of more variables, then the problem can be easily shown to encode Clique and hence become W[1]-hard (see Section 2.3). The translation entails a non-trivial analysis to ensure that all functions are indeed monotone. Specifically, each convex region requires its own set of tailored functions to enforce some relationships between the (unknown) guards it announced to contain and the (unknown) subsequences that these guards are supposed to see. In a sense, our first three components extract the algebraic essence of the Vertex-Vertex Art Gallery problem: by identifying monotone properties and making guesses to ensure binary dependencies between solution elements, the problem is encoded by a restricted constraint satisfaction problem.

Lastly, our fourth component is a relatively simple polynomial-time algorithm for Monotone 2-CSP (see Theorem 3), based on a reduction to 2-CNF-SAT. The crux is *not* to encode every pair of a variable of Monotone 2-CSP and a potential value for it as a variable of 2-CNF-SAT that signifies equality, because then, although the functions become easily encodable in the language of 2-CNF-SAT, it is unclear how to ensure that each variable of Monotone 2-CSP will be in exactly one pair that corresponds to a variable assigned true when satisfying the 2-CNF-SAT formula. Indeed, the naive approach seems futile, because it does not exploit the monotonicity of the input functions. Instead, for each pair of a variable of Monotone 2-CSP and a potential value for it with the exception of 0, we introduce a variable of 2-CNF-SAT signifying that the variable is assigned *at least* the value in the pair. The assignment of value 0 is implicitly encoded by the negation of pairs with the value 1. Then, we can ensure that each variable is assigned exactly one value (when translating a truth assignment for the 2-CNF-SAT instance we created back into an assignment for the Monotone 2-CSP input instance), and by relying on the monotonicty of the input functions, we are able to encode them correctly in the language of 2-CNF-SAT.

For notational clarity, we describe our proof for Vertex-Vertex Art Gallery. However, all arguments extend in a straightforward manner to solve the annotated generalization of Vertex-Vertex Art Gallery where G and C are each a subset of the vertex set of the polygon. Then, simple discretization procedures yield the positive resolution of the parameterized complexity also of Vertex-Boundary Art Gallery and Boundary-Vertex Art Gallery (see Section 5 of the full version [4]).

Preliminaries. We use the abbreviation Art Gallery to refer to Vertex-Vertex Art Gallery. We model a polygon by a graph $P = (V, E)$ with $V = \{1, 2, \ldots, n\}$ and $E = \{\{i, i+1\}\} : i \in \{1, \ldots, n-1\}\} \cup \{\{n, 1\}\}$. For a simple polygon P, we consider the boundary of P as part of its interior. We slightly abuse notation and refer to vertices $i \in V$ where the interior angle of P at i is 180 degrees as convex vertices. We denote the set of reflex vertices of P by $\mathsf{reflex}(P)$, and the set of convex vertices of P by $\mathsf{convex}(P)$. Given a non-convex polygon $P = (V, E)$, we suppose w.l.o.g. that $1 \in V$ is a reflex vertex. We say that a point p *sees* (or is *visible* to) a point q if every point of the line segment $\overline{pq}$ belongs to the interior of P. More generally, a set of points S *sees* a set of points Q if every point in Q is seen by at least one point in S. The definition of a convex polygon asserts the following.

▶ **Observation 4.** *Any point within a convex polygon P sees all points within P.*

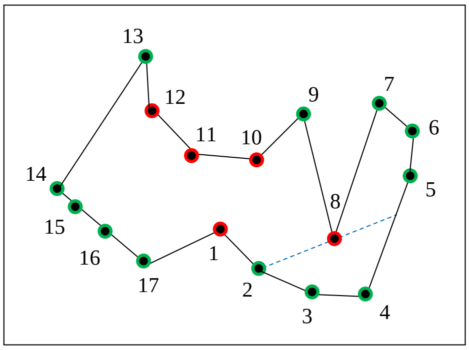

Figure 3 A simple polygon with three maximal convex regions: $[2,7]$, $[9]$ and $[13,17]$. Although $2,5 \in [2,7]$ belong to the same convex region, they do not see each other.

2 Algorithm for Art Gallery

In this section, we prove that Art Gallery is FPT with respect to r, the number of reflex vertices, by developing an algorithm with running time $2^{\mathcal{O}(r^2 \log r)} n^{\mathcal{O}(1)}$. We first present structural claims that exhibit the monotone way in which vertices in a so called "convex region" see vertices in another such region (Section 2.1). Then, we present a Turing reduction from Art Gallery to a problem called Structured Art Gallery (Section 2.2). Next, based on the claims in Section 2.1, we present our main reduction, which translates Structured Art Gallery to Monotone 2-CSP (Section 2.3). By developing an algorithm for Monotone 2-CSP, we conclude the proof.

2.1 Simple Structural Claims

We begin our analysis with the definition of a subsequence of vertices termed a convex region, illustrated in Fig. 3. Below, $j+1$ for $j = n$ refers to 1. Because we assumed that vertex 1 of any non-convex polygon is a reflex vertex, any convex region $[i,j]$ satisfies $i \neq 1$.

▶ **Definition 5.** *Let $P = (V,E)$ be a simple polygon. A non-empty set of vertices $[i,j] = \{i, i+1, \ldots, j\}$ is a* convex region *of P if all the vertices in $[i,j]$ are convex. In addition, if $i-1 \geq 1$ and $j+1$ are reflex vertices, then $[i,j]$ is a* maximal *convex region.*

In what follows, we would like to argue that for every two (not necessarily distinct) convex regions, one convex region sees the other in a manner that is "monotone" for each "orientation" in which we traverse these regions. To formalize this, we make use of the following notation, illustrated in Fig. 4. For a polygon $P = (V,E)$, a convex region $[i,j]$ of P and a vertex $v \in V$, denote the smallest and largest vertices in $[i,j]$ that are seen by v by $\mathsf{first}(v,[i,j])$ and $\mathsf{last}(v,[i,j])$, respectively. If v sees no vertex in $[i,j]$, define $\mathsf{first}(v,[i,j]) = \mathsf{last}(v,[i,j]) = \mathsf{nil}$. Accordingly, we define two types of monotone views. First, we address the orientation corresponding to first (see Fig. 4). Roughly speaking, we say that the way a convex region $[i,j]$ views a convex region $[i',j']$ is, say, non-decreasing with respect to first, if when we traverse $[i,j]$ from i to j and consider the first vertices in $[i',j']$ that vertices in $[i,j]$ see, then the sequence of these first vertices (viewed as integers) is a monotonically non-decreasing sequence once we omit all occurrences of nil from it.[3] We further demand that, between two equal vertices in this sequence, no nil occurs. Formally,

[3] A non-decreasing function (or sequence) is one that *never* decreases but can sometimes *not* increase.

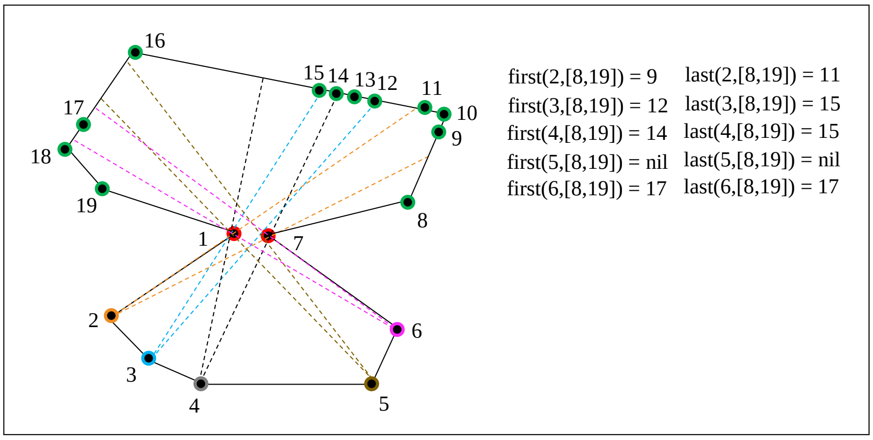

Figure 4 The way $[2,6]$ views $[8,19]$ is non-decreasing with respect to both first and last.

▶ **Definition 6.** *Let $P = (V, E)$ be a simple polygon. We say that the way a convex region $[i, j]$ of P views a (not necessarily distinct) convex region $[i', j']$ of P is* non-decreasing *(resp.* non-increasing*) with respect to* first *if for all $t, \widehat{t} \in \{i, i+1, \ldots, j\}$ such that $t \leq \widehat{t}$, $\mathsf{first}(t, [i', j']) \neq \mathsf{nil}$ and $\mathsf{first}(\widehat{t}, [i', j']) \neq \mathsf{nil}$, we have that*

- *$\mathsf{first}(t, [i', j']) \leq \mathsf{first}(\widehat{t}, [i', j'])$ (resp. $\mathsf{first}(t, [i', j']) \geq \mathsf{first}(\widehat{t}, [i', j'])$), and*
- *if $\mathsf{first}(t, [i', j']) = \mathsf{first}(\widehat{t}, [i', j'])$, then for all $p \in \{t, \ldots, \widehat{t}\}$, $\mathsf{first}(p, [i', j']) = \mathsf{first}(t, [i', j'])$.*[4]

Symmetrically, we address the orientation corresponding to the notation last.

▶ **Definition 7.** *Let $P = (V, E)$ be a simple polygon. We say that the way a convex region $[i, j]$ of P views a (not necessarily distinct) convex region $[i', j']$ of P is* non-decreasing *(resp.* non-increasing*) with respect to* last *if for all $t, \widehat{t} \in \{i, i+1, \ldots, j\}$ such that $t \leq \widehat{t}$, $\mathsf{last}(t, [i', j']) \neq \mathsf{nil}$ and $\mathsf{last}(\widehat{t}, [i', j']) \neq \mathsf{nil}$, we have that*

- *$\mathsf{last}(t, [i', j']) \leq \mathsf{last}(\widehat{t}, [i', j'])$ (resp. $\mathsf{last}(t, [i', j']) \geq \mathsf{last}(\widehat{t}, [i', j'])$), and*
- *if $\mathsf{last}(t, [i', j']) = \mathsf{last}(\widehat{t}, [i', j'])$, then for all $p \in \{t, \ldots, \widehat{t}\}$, $\mathsf{last}(p, [i', j']) = \mathsf{last}(t, [i', j'])$.*

The main purpose of this section is to prove the following two lemmas. We believe that some arguments required to establish their proofs might be folklore. The first lemma asserts that the subsequence seen by a vertex within a convex region does not contain "gaps".

▶ **Lemma 8 (♠).** *Let $P = (V, E)$ be a simple polygon, $v \in V$, and $[i, j]$ be a convex region of P. Then, v sees every vertex $t \in [i, j]$ such that $\mathsf{first}(v, [i, j]) \leq t \leq \mathsf{last}(v, [i, j])$.*[5]

The second lemma asserts that views are monotone. Intuitively, whenever we move along a convex region $[i, j]$ while viewing a convex region $[i', j']$ as described earlier, the first vertices (and last vertices) seen form a non-increasing or non-decreasing sequence.[6]

▶ **Lemma 9 (♠).** *Let $P = (V, E)$ be a simple polygon, and let $[i, j]$ and $[i', j']$ be two (not necessarily distinct) maximal convex regions of P. Then,* (i) *the way in which $[i, j]$ views $[i', j']$ with respect to first is either non-decreasing or non-increasing, and* (ii) *the way in which $[i, j]$ views $[i', j']$ with respect to last is either non-decreasing or non-increasing.*

---

[4] We remark that this condition cannot be replaced by "for all $p \in \{t, \ldots, \widehat{t}\}$, $\mathsf{first}(p, [i', j']) \neq \mathsf{nil}$". For example, in Fig. 4, neither $\mathsf{first}(4, [8, 19])$ nor $\mathsf{first}(6, [8, 19])$ is nil, but $\mathsf{first}(5, [8, 19]) = \mathsf{nil}$.

[5] If v does not see any vertex in $[i, j]$, the claim holds vacuously.

[6] We remark that we do not know whether it is possible that the first vertices would form a non-increasing (or non-decreasing) sequence and the last vertices would not. Our weaker claim suffices for our purposes.

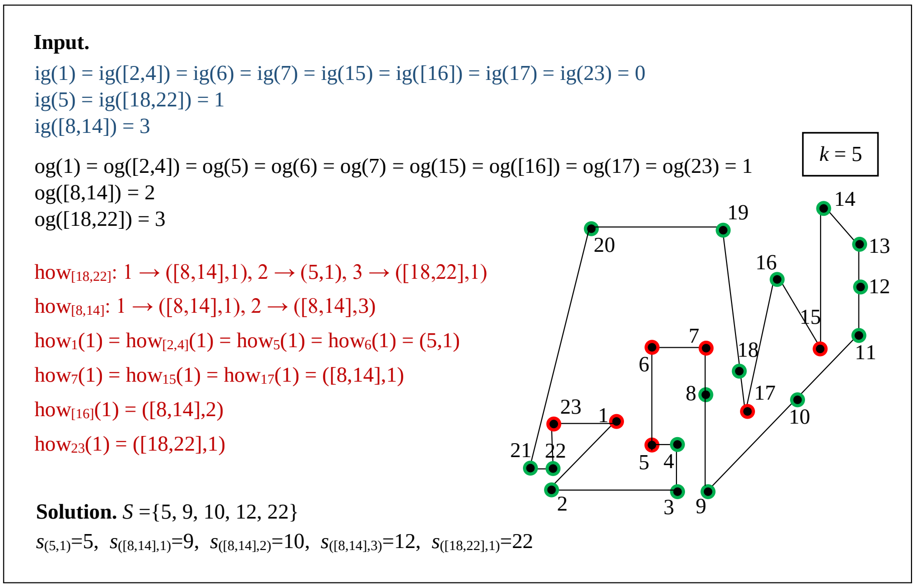

Figure 5 An input and a solution for the Structured Art Gallery problem.

2.2 Turing Reduction to Structured Art Gallery

An intermediate step in our reduction from Art Gallery to Monotone 2-CSP addresses an annotated version of Art Gallery, called Structured Art Gallery. Intuitively, in Structured Art Gallery each convex region "announces" how many guards it should contain, and how many guards are to be used to see it completely. In addition, each convex region announces by which *unknown guard* (identified as "the i^{th} guard to be placed on region C" for some i and C) its prefix should be guarded, by which unknown guard a region after this prefix should be guarded, and so on. In what follows, we formally define the Structured Art Gallery problem; then, we present our reduction from Art Gallery to Structured Art Gallery, and afterwards argue that this reduction is correct. For a polygon P, let $\mathcal{C}(P)$ be the set of maximal convex regions of P. Note that $|\mathcal{C}(P)| \leq r$.

Problem Definition. The input of Structured Art Gallery consists of a simple polygon $P = (V, E)$, a non-negative integer $k < r$, and the following functions (see Fig. 5).

- $\mathsf{ig} : \mathcal{C}(P) \cup \mathsf{reflex}(P) \to \{0, \ldots, k\}$, where $\sum_{x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)} \mathsf{ig}(x) \leq k$. Intuitively, for a convex region or reflex vertex x, ig assigns the number of guards to be placed in x.
- $\mathsf{og} : \mathcal{C}(P) \cup \mathsf{reflex}(P) \to \{1, \ldots, k\}$, where for all $x \in \mathsf{reflex}(P)$, $\mathsf{og}(x) = 1$. Intuitively, for a convex region or reflex vertex x, og assigns the number of guards required to see x.
- For each $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$, $\mathsf{how}_x : \{1, \ldots, \mathsf{og}(x)\} \to (\mathcal{C}(P) \cup \mathsf{reflex}(P)) \times \{1, \ldots, k\}$, where for each (y, i) in the image of how_x, $i \leq \mathsf{ig}(y)$. Intuitively, for any $j \in \{1, \ldots, \mathsf{og}(x)\}$, $\mathsf{how}_x(j) = (y, i)$ indicates that the j^{th} guard required to see x is the i^{th} guard placed in y.

The objective of Structured Art Gallery is to determine whether there exists a set $S \subseteq V$ of size at most k such that the following conditions hold:

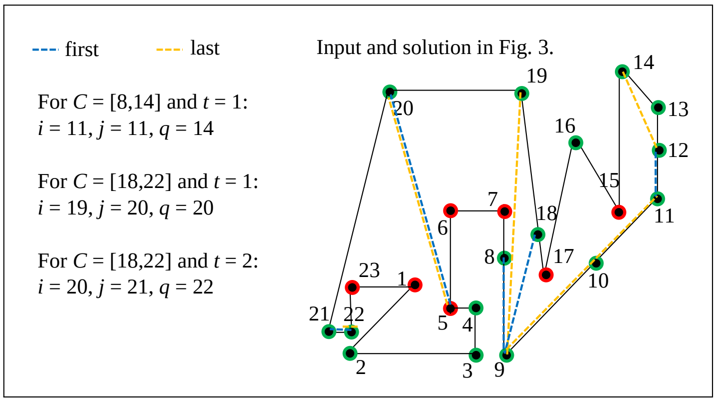

Figure 6 Condition 3b satisfied by a solution for STRUCTURED ART GALLERY.

1. For each $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$, $|S \cap x| = \mathsf{ig}(x)$.[7] Accordingly, for each $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$ and $i \in \{1, \ldots, \mathsf{ig}(x)\}$, let $s_{(x,i)}$ denote the i^{th} largest vertex in $S \cap x$ (see Fig. 5).
2. For each $x \in \mathsf{reflex}(P)$, $s_{\mathsf{how}_x(1)}$ sees x.
3. For each $C \in \mathcal{C}(P)$, the following conditions hold:
   a. $\mathsf{first}(s_{\mathsf{how}_C(1)}, C)$ is the smallest vertex in C.
   b. For every $t \in \{1, \ldots, \mathsf{og}(C) - 1\}$, denote $i = \mathsf{last}(s_{\mathsf{how}_C(t)}, C)$, $j = \mathsf{first}(s_{\mathsf{how}_C(t+1)}, C)$ and $q = \mathsf{last}(s_{\mathsf{how}_C(t+1)}, C)$. Then, *(i)* $i \geq j - 1$, and *(ii)* $i \leq q - 1$. (See Fig. 6.)
   c. $\mathsf{last}(s_{\mathsf{how}_C(\mathsf{og}(C))}, C)$ is the largest vertex in C.

Informally, Condition 3b states that *(i)* the last vertex in C seen by its t^{th} guard should be at least as large as the predecessor of the first vertex in C seen by its $(t+1)^{th}$ guard, and *(ii)* the last vertex in C seen by its t^{th} guard should be smaller than the last vertex in C seen by its $(t+1)^{th}$ guard. The first condition ensures that no unseen "gaps" are created within C, while the second condition ensures that as the index t grows larger, the last vertex seen by the t^{th} guard grows larger as well. (The second condition will be part of our transition towards the interpretation of the objective of ART GALLERY by *binary* constraints.)

Turing Reduction. Given an instance (P, k) of ART GALLERY, in case $r \leq k$, output YES.[8] Otherwise, the output of the reduction, $\mathsf{reduction}(P, k)$, is the set of all instances $(P, k, \mathsf{ig}, \mathsf{og}, \{\mathsf{how}_x\}|_{x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)})$ of STRUCTURED ART GALLERY.

Observe that $|\mathcal{C}(P) \cup \mathsf{reflex}(P)| \leq 2r$, and therefore the number of possible functions ig is upper bounded by $(k+1)^{2r}$, the number of possible functions og is upper bounded by k^{2r}, and for each $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$, the number of possible functions how_x is upper bounded by $(2rk)^k$. Hence, the number of instances produced is upper bounded by $(k+1)^{2r} \cdot k^{2r} \cdot ((2rk)^k)^{2r}$. When $k \leq r$, this number is upper bounded by $r^{\mathcal{O}(r^2)}$. Moreover, the instances in $\mathsf{reduction}(P, k)$ can be enumerated with polynomial delay. Thus,

▶ **Observation 10.** *Let (P, k) be an instance of ART GALLERY. Then, $|\mathit{reduction}(P, k)| = r^{\mathcal{O}(r^2)}$, and $\mathit{reduction}(P, k)$ is computable in time $r^{\mathcal{O}(r^2)} n^{\mathcal{O}(1)}$.*

[7] If $x \in \mathsf{reflex}(P)$, by $S \cap x$ we mean $S \cap \{x\}$.
[8] To comply with the formal definition of a Turing reduction, by YES we mean a set with a single trivial YES-instance of STRUCTURED ART GALLERY.

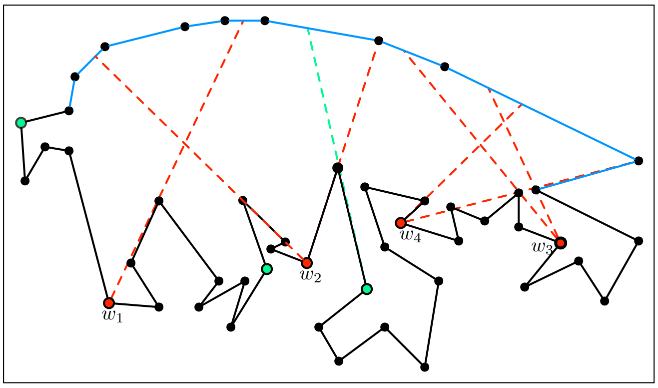

Figure 7 Example of a possible selection of $w_1, w_2, \ldots, w_p$. Solution vertices are colored green and red, and C is colored blue.

Correctness. Our proof of correctness crucially relies on Lemma 8 and Proposition 1.

▶ **Lemma 11.** *An instance (P, k) is a* YES*-instance of* ART GALLERY *if and only if there is a* YES*-instance of* STRUCTURED ART GALLERY *in* reduction(P, k).

Proof.

Forward Direction. Suppose that (P, k) is a YES-instance of ART GALLERY and that $r > k$. Accordingly, let $S \subseteq V$ be a solution to (P, k). We first define the function $\mathsf{ig} : \mathcal{C}(P) \cup \mathsf{reflex}(P) \to \{0, \ldots, k\}$ as follows. For each $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$, let $\mathsf{ig}(x) = |S \cap x|$. Because $|S| \leq k$ (since S is a solution to (P, k)), we have that $\sum_{x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)} \mathsf{ig}(x) \leq k$. For each $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$, we order the vertices in $S \cap x$ from smallest to largest, and denote them accordingly by $s_{(x,1)}, s_{(x,2)}, \ldots, s_{(x,\mathsf{ig}(x))}$.

We define the functions $\mathsf{og} : \mathcal{C}(P) \cup \mathsf{reflex}(P) \to \{1, \ldots, k\}$ and $\mathsf{how}_x : \{1, \ldots, \mathsf{og}(x)\} \to (\mathcal{C}(P) \cup \mathsf{reflex}(P)) \times \{1, \ldots, k\}$ for all $x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)$. For each reflex vertex $x \in \mathsf{reflex}(P)$, define $\mathsf{og}(x) = 1$, and $\mathsf{how}_x(1) = (y, i)$ for some vertex $s_{(y,i)} \in S$ that sees x. The existence of such a vertex $s_{(y,i)}$ follows from the assertion that S is a solution to (P, k). For each convex region $C \in \mathcal{C}(P)$, define $\mathsf{og}(C)$ and how_C as follows. Let W denote the set of vertices in S that see at least one vertex in C. Since W sees C, there exists a vertex in W that sees the smallest vertex in C. Pick such a vertex arbitrarily and denote it by w_1. Now, if w_1 does not see the largest vertex in C, then there exists a vertex in W that sees the smallest vertex in C that is larger than the largest vertex seen by w_1. We pick such a vertex arbitrarily, and denote it by w_2. Next, if w_2 does not see the largest vertex in C, then there exists a vertex in W that sees the smallest vertex in C that is larger than the largest vertex seen by w_2. We pick such a vertex arbitrarily, and denote it by w_3. Similarly, we define $w_4, w_5, \ldots, w_p$, for the appropriate $p \in \{1, \ldots, k\}$ (see Fig. 7). Here, the supposition that $p \leq k$ follows from Lemma 8, which implies that $w_i \neq w_j$ for all distinct $i, j \in \{1, \ldots, p\}$. We define $\mathsf{og}(C) = p$, and for all $t \in \{1, \ldots, \mathsf{og}(C)\}$, we define $\mathsf{how}_C(t) = (y, i)$ for the pair $(y, i) \in (\mathcal{C}(P) \cup \mathsf{reflex}(P)) \times \{1, \ldots, k\}$ that satisfies $w_t = s_{(y,i)}$.

Our definitions directly ensure that for each $C \in \mathcal{C}(P)$, the following conditions hold:

1. $\mathsf{first}(s_{\mathsf{how}_C(1)}, C)$ is the smallest vertex in C.
2. For every $t \in \{1, \ldots, \mathsf{og}(C) - 1\}$, denote $i = \mathsf{last}(s_{\mathsf{how}_C(t)}, C)$, $j = \mathsf{first}(s_{\mathsf{how}_C(t+1)}, C)$ and $q = \mathsf{last}(s_{\mathsf{how}_C(t+1)}, C)$. Then, *(i)* $i \geq j - 1$, and *(ii)* $i \leq q - 1$.
3. $\mathsf{last}(s_{\mathsf{how}_C(\mathsf{og}(C))}, C)$ is the largest vertex in C.

By the arguments above, $I = (P, k, \mathsf{ig}, \mathsf{og}, \{\mathsf{how}_x\}|_{x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)})$ is an instance of Structured Art Gallery, and S is a solution to I. Since $I \in \mathsf{reduction}(P, k)$, the proof of the forward direction is complete.

Reverse Direction. If $k \geq r$, then we output Yes (or rather a trivial Yes-instance), and by Proposition 1, indeed the input is a Yes-instance as well. Next, suppose that $k < r$, and there is a Yes-instance $I = (P, k, \mathsf{ig}, \mathsf{og}, \{\mathsf{how}_x\}|_{x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)})$ in $\mathsf{reduction}(P, k)$. Accordingly, let $S \subseteq V$ be a solution to I. Then, $|S| \leq k$. Thus, to prove that (P, k) is a Yes-instance of Art Gallery, it suffices to show that S sees V. For each $x \in \mathsf{reflex}(P)$, $s_{\mathsf{how}_x(1)}$ sees x, and therefore S sees $\mathsf{reflex}(P)$.

Now, we show that S sees $\mathsf{convex}(P)$. To this end, we choose a convex region $[i, j] \in \mathcal{C}(P)$, and show that S sees $[i, j]$. Specifically, for each $p \in \{i, \ldots, j\}$, we prove that there is $t \in \{1, \ldots, \mathsf{og}([i, j])\}$ such that $s_{\mathsf{how}_{[i,j]}(t)}$ (which is a vertex in S) sees p. The proof is by induction on p. In the basis, where $p = i$, correctness follows from the assertion that $\mathsf{first}(s_{\mathsf{how}_{[i,j]}(1)}, [i, j])$ is the smallest vertex in $[i, j]$. Now, we suppose that the claim is correct for p, and prove it for $p + 1$. By the inductive hypothesis, there is $t \in \{1, \ldots, \mathsf{og}([i, j])\}$ such that $s_{\mathsf{how}_{[i,j]}(t)}$ sees p. If $s_{\mathsf{how}_{[i,j]}(t)}$ sees $p + 1$, then we are done. Thus, we now suppose that $s_{\mathsf{how}_{[i,j]}(t)}$ does not see $p + 1$. Then, $\mathsf{last}(s_{\mathsf{how}_{[i,j]}(t)}, [i, j]) = p$. We have two cases:

- First, consider the case where $t < \mathsf{og}([i, j])$. Then, because S is a solution to I, the vertex $p = \mathsf{last}(s_{\mathsf{how}_{[i,j]}(t)}, [i, j])$ is larger or equal to $d - 1$ for $d = \mathsf{first}(s_{\mathsf{how}_{[i,j]}(t+1)}, [i, j])$. This means that $\mathsf{first}(s_{\mathsf{how}_{[i,j]}(t+1)}, [i, j]) \leq p + 1$. Moreover, p is smaller than the vertex $\mathsf{last}(s_{\mathsf{how}_{[i,j]}(t+1)}, [i, j])$. Thus, $p + 1 \leq \mathsf{last}(s_{\mathsf{how}_{[i,j]}(t+1)}, [i, j])$. Then, $\mathsf{first}(s_{\mathsf{how}_{[i,j]}(t+1)}, [i, j]) \leq p + 1 \leq \mathsf{last}(s_{\mathsf{how}_{[i,j]}(t+1)}, [i, j])$. By Lemma 8, $s_{\mathsf{how}_{[i,j]}(t+1)}$ sees $p + 1$.
- Second, consider the case where $t = \mathsf{og}([i, j])$. In this case, because S is a solution to I, we have that $\mathsf{last}(s_{\mathsf{how}_{[i,j]}(\mathsf{og}([i,j]))}, [i, j])$ is the largest vertex in $[i, j]$. Thus, $p + 1 \leq \mathsf{last}(s_{\mathsf{how}_{[i,j]}(\mathsf{og}([i,j]))}, [i, j])$, which is a contradiction. ◀

2.3 Karp Reduction to Monotone 2-CSP

We proceed to the second part of our proof, a reduction from Structured Art Gallery to Monotone 2-CSP.[9] (The analysis of this can be found in the full version of the paper [4]).

Problem Definition. The input of Monotone 2-CSP consists of a set X of *variables*, denoted by $X = \{x_1, x_2, \ldots, x_{|X|}\}$, a set C of *constraints*, and $N \in \mathbb{N}$ given in unary. Each constraint $c \in C$ has the form $[x_i \,\mathsf{sign}\, f(x_j)]$ where $i, j \in \{1, \ldots, |X|\}$, $\mathsf{sign} \in \{\geq, \leq\}$ and $f : \{0, \ldots, N\} \to \{0, \ldots, N\}$ is a monotone function. An assignment $\alpha : X \to \{0, \ldots, N\}$ *satisfies* a constraint $c = [x_i \,\mathsf{sign}\, f(x_j)] \in C$ if $[\alpha(x_i) \,\mathsf{sign}\, f(\alpha(x_j))]$ is true. The objective of Monotone 2-CSP is to decide if there exists an assignment $\alpha : X \to \{0, \ldots, N\}$ that satisfies all the constraints in C (see Fig. 8).

If the function f of a constraint $c = [x_i \,\mathsf{sign}\, f(x_j)]$ is constantly β (that is, for every $t \in \{0, \ldots, N\}$, $f(t) = \beta$), then we use the shorthand $c = [x_i \,\mathsf{sign}\, \beta]$. Moreover, we suppose that every constraint represented by a quadruple is associated with two distinct variables.

Karp Reduction. Given an instance $I = (P, k, \mathsf{ig}, \mathsf{og}, \{\mathsf{how}_x\}|_{x \in \mathcal{C}(P) \cup \mathsf{reflex}(P)})$ of Structured Art Gallery, define an instance $\mathsf{reduction}(I) = (X, C, N)$ of Monotone 2-CSP as follows. Let $k^\star = \sum_{e \in \mathcal{C}(P) \cup \mathsf{reflex}(P)} \mathsf{ig}(e)$, $X = \{x_1, x_2, \ldots, x_{k^\star}\}$ and $N = n + 1$. (Here,

[9] CSP is an abbreviation of Constraint Satisfaction Problem, and 2 is the maximum arity of a constraint.

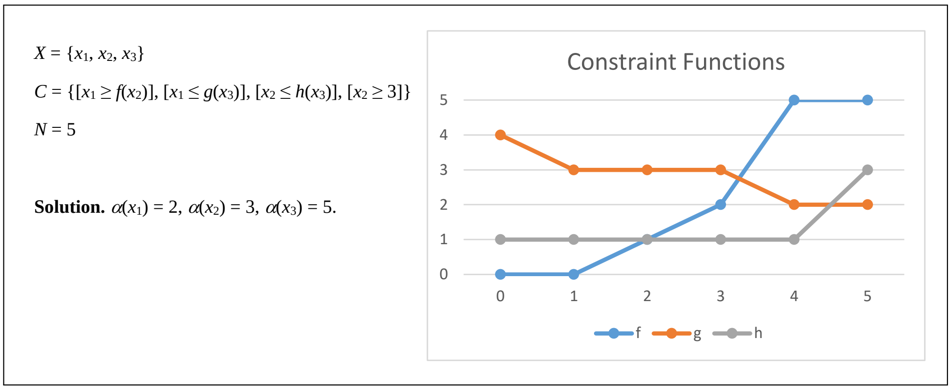

Figure 8 An input for MONOTONE 2-CSP that has a unique solution.

$n = |V|$.) Additionally, let bij be an arbitrary bijective function from X to $\{(e,i) : e \in \mathcal{C}(P) \cup \mathsf{reflex}(P), i \in \{1,\dots,\mathsf{ig}(e)\}\}$. Intuitively, for any variable $x \in X$ with $\mathsf{bij}(x) = (e,i)$, we think of x as the i^{th} guard to be placed in region e. In particular, the value to be assigned to x is the identity of this guard. The values 0 and $n+1$ are not identities of vertices in V, and we will ensure that no solution assignment assigns them; we note that these two values are useful because they will allow us to exclude assignments that should not be solutions. Next, we define our constraints and show that their functions are monotone.

Association. For each $x \in X$ with $\mathsf{bij}(x) = (e,i)$, we need to ensure that the vertex assigned to x is within the region e. To this end, we introduce the following constraints.

- If $e \in \mathsf{reflex}(P)$, then insert the constraint $[x = e]$. (That is, insert $[x \leq e]$ and $[x \geq e]$.)
- Else, $\mathsf{bij}(x) = (e,j)$ for $e \in \mathcal{C}(P)$. Let ℓ and h be the smallest and largest vertices in e, respectively, and insert the constraints $[x \geq \ell]$ and $[x \leq h]$.

Let A denote this set of constraints.

Order in a convex region. For all $x, x' \in X$ where $\mathsf{bij}(x) = (C,i)$ and $\mathsf{bij}(x') = (C,j)$ for the same convex region $C \in \mathcal{C}(P)$ and $i < j$, we need to ensure that the vertex assigned to x' is larger than the one assigned to x. To this end, we introduce the constraint $[x' \geq f(x)]$ where f is defined as follows. For all $q \in \{0,\dots,N-1\}$, $f(q) = q+1$, and $f(N) = N$. Let O denote this set of constraints. We note that the constraints in $A \cup O$ together enforce each variable $x \in X$ with $\mathsf{bij}(x) = (C,i)$ for $C \in \mathcal{C}(P)$ to be assigned the i^{th} guard placed in C.

Guarding reflex vertices. For every reflex vertex $y \in \mathsf{reflex}(P)$ with $\mathsf{how}_y(1) = (e,i)$, we need to ensure that the vertex assigned to $x = \mathsf{bij}^{-1}(e,i)$ sees y. To this end, consider two cases. First, suppose that $e \in \mathsf{reflex}(P)$. Then, *(i)* if e does not see y, output NO, and *(ii)* else, no constraint is introduced. Second, suppose that $e \in \mathcal{C}(P)$. Denote $\ell = \mathsf{first}(y,e)$ and $h = \mathsf{last}(y,e)$. Then, *(i)* if ℓ (and thus also h) is nil, then output NO, and *(ii)* else, introduce the constraints $c_y^1 = [x \geq \ell]$ and $c_y^2 = [x \leq h]$.

Guarding first vertices in convex regions. For every convex region $C = [q,q'] \in \mathcal{C}(P)$ with $\mathsf{how}_C(1) = (e,i)$, we need to ensure that the vertex assigned to $x = \mathsf{bij}^{-1}(e,i)$ sees q, the first vertex of C. To this end, consider two cases. First, suppose that $e \in \mathsf{reflex}(P)$. Then, *(i)* if e does not see q, output NO, and *(ii)* else, no constraint is introduced. Second, suppose that $e \in \mathcal{C}(P)$. Denote $\ell = \mathsf{first}(q,e)$ and $h = \mathsf{last}(q,e)$. Then, *(i)* if ℓ is nil, then output NO, and *(ii)* else, insert the constraints $c^1_{(C,1)} = [x \geq \ell]$ and $c^2_{(C,1)} = [x \leq h]$.

Guarding last vertices in convex regions. For every convex region $C = [q, q'] \in \mathcal{C}(P)$ with $\mathsf{how}_C(\mathsf{og}(C)) = (e, i)$, we need to ensure that the vertex assigned to $x = \mathsf{bij}^{-1}(e, i)$ sees q', the last vertex of C. To this end, consider two cases. First, suppose that $e \in \mathsf{reflex}(P)$. Then, *(i)* if e does not see q', output No, and *(ii)* else, no constraint is introduced. Second, suppose that $e \in \mathcal{C}(P)$. Denote $\ell = \mathsf{first}(q', e)$ and $h = \mathsf{last}(q', e)$. Then, *(i)* if ℓ is nil, then output No, and *(ii)* else, insert the constraints $c^1_{(C,\mathsf{og}(C))} = [x \geq \ell]$ and $c^2_{(C,\mathsf{og}(C))} = [x \leq h]$.

Guarding middle vertices in convex regions. For every convex region $C \in \mathcal{C}(P)$ and $t \in \{2, \ldots, \mathsf{og}(C)\}$, we introduce four constraints based on the following notation.

- $(e, \gamma) = \mathsf{how}_C(t)$ and $x = \mathsf{bij}^{-1}(e, \gamma)$. Intuitively, the t^{th} vertex to guard C should be the γ^{th} guard to be placed in e, and its precise identity should be assigned to x. If no vertex in e sees at least one vertex in C, then return No.[10] Let ℓ and h be the smallest and largest vertices in e that see at least one vertex in C, respectively.
- $(e', \gamma') = \mathsf{how}_C(t-1)$ and $x' = \mathsf{bij}^{-1}(e', \gamma')$. Intuitively, the $(t-1)^{\text{th}}$ vertex to guard C should be the ${\gamma'}^{\text{th}}$ guard to be placed in e', and its precise identity should be assigned to x'. If no vertex in e' sees at least one vertex in C, then return No. Let ℓ' and h' be the smallest and largest vertices in e' that see at least one vertex in C, respectively.

Now, insert the constraints $\widetilde{c}^1_{(C,t)} = [x \geq \ell]$ and $\widetilde{c}^2_{(C,t)} = [x \leq h]$. Intuitively, these two constraints *help* to ensure that x will be assigned a vertex that sees at least one vertex in C. However, these constraints alone are insufficient for this task – ensuring that we pick a guard between two vertices that see vertices in C does not ensure that this guard sees vertices in C.[11] Nevertheless, combined with our final constraints, this task is achieved.

Lastly, we consider two sets of four cases. The first set introduces a constraint to ensure that x, which stands for the t^{th} vertex to guard C, should satisfy that the first vertex in C seen by x is smaller or equal than the vertex larger by 1 than the last vertex in C seen by x', which stands for the $(t-1)^{\text{th}}$ vertex to guard C. On the other hand, the second set introduces a constraint to ensure that the last vertex in C seen by x is larger than the last vertex in C seen by x'. Together, because views have no "gaps", this would imply that x sees the vertex in C that is larger by 1 than the last vertex in C seen by x'. Due to lack of space, we only present the first case of each set. (Omitted details can be found in the full version [4]). To unify notation, if e (or e') is a reflex vertex, we say that the way e (or e') views C is non-decreasing with respect to both first and last.

First, consider the case where the way e' views C is non-decreasing with respect to last, and the way e views C is non-decreasing with respect to first. We insert a constraint $[x \leq f(x')]$, where f (having domain and range $\{0, \ldots, N\}$) is defined as follows.

- For all $i < \ell'$: $f(i) = 0$. Intuitively, we forbid x to be assigned a vertex smaller than the first vertex in e that can see C.
- For $i = \ell', \ell' + 1, \ldots, h'$: Denote $a = \mathsf{last}(i, C)$. We have two subcases.
  - If **(i)** $a = \mathsf{nil}$, **(ii)** $a + 1 \notin C$, or **(iii)** $\mathsf{first}(j, C) \leq a + 1$ for no $j \in e$, let $f(i) = f(i-1)$. Roughly speaking, given that x' sees C, $a \neq \mathsf{nil}$ (in cases we will care about). Moreover, $a + 1 \in C$ will be ensured by the second set of cases and the way we guard the last vertex of a convex region. Lastly, $\mathsf{first}(j, C) \leq a + 1$ for some $j \in e$ will be ensured using that $f(i-1)$ (unless $f(i-1) = 0$) is a vertex that sees $a + 1$.

[10] In case $e \in \mathsf{reflex}(P)$, we mean that e itself does not see any vertex in C.
[11] For example, in Fig. 4, neither $\mathsf{first}(4, [8, 19])$ nor $\mathsf{first}(6, [8, 19])$ is nil, but $\mathsf{first}(5, [8, 19]) = \mathsf{nil}$.

  - Else, let j be the largest vertex in e such that $\mathsf{first}(j, C) \leq a + 1$. Define $f(i) = j$. Intuitively, by enforcing x to be smaller or equal than j – the largest vertex in e that might see $a + 1$ – we ensure that the following condition holds: the first vertex x sees in C, under the assumption that it is not nil,[12] is smaller or equal to $a + 1$ (because the way e views C is non-decreasing with respect to first).
- For all $i > h'$: $f(i) = N$.

Second, consider the case where the ways e' and e view C are both non-decreasing with respect to last. We insert a constraint $[x \geq f(x')]$, where f is defined as follows.
- For all $i > h'$: $f(i) = N$.
- For $i = h', h' - 1, \ldots, \ell'$: Denote $a = \mathsf{last}(i, C)$. We have two subcases.
  - If **(i)** $a = \mathsf{nil}$, **(ii)** $a + 1 \notin C$, or **(iii)** $\mathsf{last}(j, C) \geq a + 1$ for no $j \in e$, let $f(i) = f(i + 1)$.
  - Else, let j be the smallest vertex in e such that $\mathsf{last}(j, C) \geq a + 1$. Define $f(i) = j$.
- For all $i < \ell'$: $f(i) = 0$.

Here, as the sign is $\geq$ and f is monotonically non-decreasing, f must be defined first for N, then for $N - 1$, and so on. Then, as long as i is such that $\mathsf{last}(j, C) \geq a + 1$ for no $j \in e$ (a case that we want to avoid), $f(i) = N$ and hence $[x \geq f(i)]$ cannot be satisfied.

References

1. Mikkel Abrahamsen, Anna Adamaszek, and Tillmann Miltzow. Irrational guards are sometimes needed. In *Proceedings of the 33rd International Symposium on Computational Geometry (SoCG)*, pages 3:1–3:15, 2017. `doi:10.4230/LIPIcs.SoCG.2017.3`.
2. Mikkel Abrahamsen, Anna Adamaszek, and Tillmann Miltzow. The art gallery problem is $\exists\mathbb{R}$-complete. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 65–73, 2018.
3. Alok Aggarwal. *The art gallery theorem: its variations, applications and algorithmic aspects*. PhD thesis, The Johns Hopkins University, Baltimore, Maryland, 1986.
4. Akanksha Agrawal, Kristine V. K. Knudsen, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. The parameterized complexity of guarding almost convex polygons, 2020. `arXiv:2003.07793`.
5. Édouard Bonnet and Tillmann Miltzow. Parameterized hardness of art gallery problems. In *Proceedings of the 24th Annual European Symposium on Algorithms (ESA)*, pages 19:1–19:17, 2016.
6. Vasek Chvátal. A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory, Series B*, 18(1):39–741, 1975.
7. Pedro J. de Rezende, Cid C. de Souza, Stephan Friedrichs, Michael Hemmer, Alexander Kröller, and Davi C. Tozoni. Engineering art galleries. In *Algorithm Engineering - Selected Results and Surveys*, pages 379–417. Springer, 2016.
8. Steve Fisk. A short proof of Chvátal's watchman theorem. *Journal of Combinatorial Theory, Series B*, 24(3):374, 1978.
9. Subir Kumar Ghosh and Partha P. Goswami. Unsolved problems in visibility graphs of points, segments, and polygons. *ACM Computing Surveys*, 46(2):22:1–22:29, 2013.
10. Panos Giannopoulos. Open problems: Guarding problems. *Lorentz Workshop on Fixed-Parameter Computational Geometry, Leiden, the Netherlands*, page 12, 2016.
11. Matthew J Katz and Gabriel S Roisman. On guarding the vertices of rectilinear domains. *Computational Geometry*, 39(3):219–228, 2008.

[12] In the proof, to ensure that this vertex is indeed not nil, we will utilize both sets of cases, together with $\widetilde{c}^1_{(C,t)}$ and $\widetilde{c}^2_{(C,t)}$, to argue that x is between two vertices seen by $a + 1$ and hence must see $a + 1$ itself.

12 D. T. Lee and Arthur K. Lin. Computational complexity of art gallery problems. *IEEE Transactions on Information Theory*, 32(2):276–282, 1986.

13 Rolf Niedermeier. Ubiquitous parameterization - invitation to fixed-parameter algorithms. In *Proceedings of the 29th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, pages 84–103, 2004.

14 Rolf Niedermeier. Reflections on multivariate algorithmics and problem parameterization. In *Proceedings of the 27th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 17–32, 2010.

15 Joseph O'rourke. *Art gallery theorems and algorithms*, volume 57. Oxford University Press Oxford, 1987.

16 Joseph O'Rourke and Kenneth J. Supowit. Some NP-hard polygon decomposition problems. *IEEE Transactions on Information Theory*, 29(2):181–189, 1983.

17 Dietmar Schuchardt and Hans-Dietrich Hecker. Two NP-hard art-gallery problems for ortho-polygons. *Mathematical Logic Quarterly*, 41(2):261–267, 1995.

18 Thomas C Shermer. Recent results in art galleries (geometry). *Proceedings of the IEEE*, 80(9):1384–1399, 1992.

19 Jorge Urrutia. Art gallery and illumination problems. *Handbook of computational geometry*, 1(1):973–1027, 2000.